NỘI DUNG THI HẾT MÔN LẬP TRÌNH HỢP NGỮ

1. CẤU TRÚC ĐỀ: gồm 2-3 câu
2. HÌNH THỨC THI: vấn đáp (sau khi đã thực hành lập trình trên máy)
3. THỜI GIAN: 30 phút chuẩn bị
4. NỘI DUNG:

Khái niệm cơ bản: các kiểu dữ liệu, phạm vi biểu diễn dữ liệu, cách thức biểu diến số có dấu, không dấu, chuyển đổi giữa các kiểu dữ liệu.

Kiến trúc x86: các thanh ghi, chức năng các thanh ghi, theo dõi giá trị các thanh ghi, các loại cờ, theo dõi giá trị cờ, set/clear giá trị;

Giới thiệu Ngôn ngữ lập trình Assembly: các thao tác setup tạo project, tạo template project, biên dịch dạng console, dạng windows…

Trao đổi dữ liệu, địa chỉ và các phép toán số học: các phép toán số học và việc chuyển dữ liệu từ bộ nhớ, thanh ghi, biến số với nhau. Lưu ý về kiểu dữ liệu khi thực hiện các phép toán.

Thủ tục thực thi: cách viết một procedure, gọi thủ tục thực hiện, stack (xem được nội dung và theo dõi sự thay đổi của stack)

Xử lý có điều kiện: các lệnh so sánh, nhảy để thực thi các điều kiện tương ứng

Các phép toán trên số nguyên: giải các bài toán cụ thể hoặc viết lại bằng ASM các chương trình tương ứng trên ngôn ngữ lập trình bậc cao (C/C++).

Xâu ký tự và mảng số: Cách nhập, xuất dữ liệu, quản lý bộ nhớ

Lập trình trong môi trường Windows: xem ví dụ về tạo mess.box. và thử tìm cách thay đổi một số nội dung (chú ý cách biên dịch chương trình)

ĐỀ THI MẪU

(Thời gian 30 phút chuẩn bị)

Câu 1 (3đ): Viết chương trình nhập vào 2 số nguyên 8 bits có dấu (x,y) vào các thanh ghi EAX,EBX và thực hiện tính tổng 2 số, kết quả lưu vào thanh ghi ECX. Giải thích kết quả thu được trong trường hợp x=128; y=-254.

num1 BYTE 128

num2 sBYTE -254

.code

main PROC

mov eax,0

mov ebx,0

mov ecx,0

movzx eax, num1

movsx ebx ,num2

add eax,ebx

mov ecx,eax

- Giải thích: Thanh ghi không lưu giá trị âm vào trong memory mà sử dụng Two’s Complement để biểu diễn số âm. Vì vậy, giá trị trong num2 sẽ là 02. cộng vào ra 82 thoai

Câu 2(3đ): Viết chương trình sắp xếp dãy 10 số nguyên theo thứ tự không giảm dưới dạng procedure và gọi thực thi procedure. **Theo dõi tình trạng của stack trước và sau khi procedure được gọi.**

Câu 3(4đ): Viết chương trình tạo một messagebox gửi cảnh báo tới người dùng, thông điệp cảnh báo được nhập từ bàn phím.

INCLUDE C:\Irvine\Irvine32.inc

.data

MB\_OK EQU 0

STR1 DB "Thong bao", 0

str2 DB " what the hell is goinging mondskdjsalkd", 0

HW DD ?

EXTERN MessageBoxA@16:NEAR; This is API cua HDH

.code

main PROC

mov edx, offset str2

mov ecx, 50

call readstring

PUSH MB\_OK

PUSH OFFSET STR1

PUSH OFFSET str2

PUSH HW

CALL MessageBoxA@16

INVOKE ExitProcess, 0

; exit

main ENDP

END main

NOTE:

- STR2 ban đầu khai báo phải có string length tương đương string.length nhập vào